Natural Language Generation from Pictographs

Leen Sevens, Vincent Vandeghinste, Ineke Schuurman, Frank Van Eynde
Centre for Computational Linguistics
KU Leuven, Belgium
firstname@ccl.kuleuven.be

Abstract

We present a Pictograph-to-Text translation system for people with Intellectual or Developmental Disabilities (IDD). The system translates pictograph messages, consisting of one or more pictographs, into Dutch text using WordNet links and an n-gram language model. We also provide several pictograph input methods assisting the users in selecting the appropriate pictographs.

1 Introduction

Being unable to access ICT is a major form of social exclusion. For people with IDD, the use of social media or applications that require the user to be able to read or write, such as email clients, is a huge stumbling block if no personal assistance is given. There is a need for digital communication interfaces that enable written contact for people with IDD.

Augmentative and Alternative Communication (AAC) assists people with communication disabilities to be socially active in the digital world. Pictographically augmented text is a specific form of AAC that is often used in schools, institutions, and sheltered workshops to allow accessible communication. Between two and five million people in the European Union could benefit from symbols or symbol-related text as a means of written communication (Keskinen et al., 2012).

Within the Able to Include framework,[1] a EU project aiming to improve the living conditions of people with IDD, we developed a Pictograph-to-Text translation system. It provides help in constructing Dutch textual messages by allowing the user to input a series of pictographs and translates these messages into NL. English and Spanish versions of the tool are currently in development. It can be considered as the inverse translation engine of the Text-to-Pictograph system as described by Vandeghinste et al. (Accepted), which is primarily conceived to improve *comprehension* of textual content.

The system converts Sclera[2] and Beta[3] input messages into Dutch text, using WordNet synsets and a trigram language model. After a discussion of related work (section 2), we describe some characteristics of pictograph languages (section 3), followed by an overview of the different pictograph input methods (section 4). The next part (section 5) is dedicated to the architecture. We present our preliminary results for Pictograph-to-Dutch translation in section 6. Finally, we conclude and discuss future work in section 7.

2 Related work

Our task shares elements with regular machine translation between natural languages and with Natural Language Generation (NLG). Jing (1998) retrieves the semantic concepts from WordNet and maps them to appropriate words to produce large amounts of lexical paraphrases for a specific application domain. Similar to our approach, Liu (2003) uses statistical language models as a solution to the word inflection problem, as there may exist multiple forms for a concept constituent. The language model re-scores all inflection forms in order to generate the best hypothesis in the output. Our solution is specifically tailored towards translation from pictographs into text.

A number of pictograph-based input interfaces can be found in the literature. Finch et al. (2011) developed picoTrans, a mobile application which allows users to build a source text by combining pictures or common phrases, but their application is not intended for people with cognitive disabilities. The Prothèse Vocale Intelligente (PVI) sys-

[1] http://abletoinclude.eu/

[2] http://www.sclera.be/

[3] https://www.betasymbols.com/

Proceedings of the 15th European Workshop on Natural Language Generation (ENLG), pages 71–75,
Brighton, September 2015.

tem by Vaillant (1998) offers a limited vocabulary of pictographs, each one corresponding to a single word. PVI searches for predicative elements, such as verbs, and attempts to fill its semantic slots, after which a tree structure is created and a grammatical sentence is generated. Fitrianie and Rothkrantz (2009) apply a similar method, requiring the user to first select the pictograph representation of a verb and fill in the role slots that are made available by that verb. Their system does not take into account people with cognitive disabilities. Various pictograph chat applications, such as Messenger Visual (Tuset et al., 1995) and Pictograph Chat Communicator III (Munemori et al., 2010), allow the user to insert pictographs, but they do not generate NL.

The Pictograph-to-Text translation engine differs from these applications in that it is specifically designed for people with cognitive disabilities, does not impose any limits on the way in which pictograph messages are composed and generates NL output where possible. Furthermore, the system's architecture is as language-independent as possible, making it very easy to add new target languages.

3 Pictograph languages

Many pictograph systems are in place. Although differences exist across pictograph sets, some features are shared among them. A pictograph of an entity (noun) can stand for one or multiple instances of that entity. Pictographs depicting actions (verbs) are deprived of aspect, tense, and inflection information. Auxiliaries and articles usually have no pictograph counterpart. Pictograph languages are simplified languages, often specifically designed for people with IDD. The Pictograph-to-Text translation system currently gives access to two pictograph sets, Sclera and Beta (see Figure 1).

Sclera pictographs[4] are mainly black-and-white pictographs. They often represent *complex* concepts, such as a verb and its object (such as *to feed the dog*) or compound words (such as *carrot soup*). There are hardly any pictographs for adverbs or prepositions.

The *Beta* set[5] is characterized by its overall consistency. Beta hardly contains any complex pictographs. Most of the pictographs represent *simplex* concepts.

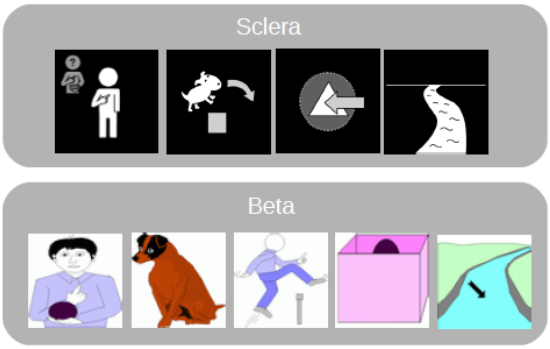

Figure 1: Example of a Beta and a Sclera sequence. Pictographs can correspond to different words and word forms in a NL, as shown for English in this example. The Sclera sequence contains a complex pictograph, namely the jumping dog.

4 Pictograph input methods

The Pictograph-to-Text translation engine relies on pictograph input and the user should be able to efficiently select the desired pictographs. We have developed two different input methods. The first approach offers a static hierarchy of pictographs, while the second option scans the user input and dynamically adapts itself in order to suggest appropriate pictographs. Usability tests will have to be performed with the target audience.

The *static hierarchy of pictographs* consists of three levels. The structure of the hierarchy is based on topic detection and frequency counts applied to 69,636 email messages sent by users of the WAI-NOT communication platform.[6]

The second method is a *dynamic pictograph prediction tool*, the first of its kind. Two different prototypes have been developed, which will eventually be merged. The first model relies on *n-gram information*. The WAI-NOT email corpus was translated into pictographs (285,372 Sclera pictographs and 284,658 Beta pictographs) in order to enable building a language model using the

[4]Freely available under Creative Commons License 2.0.

[5]The coloured pictographs can be obtained at reasonable prices. Their black-and-white equivalents are available for free.

[6]http://www.wai-not.be/ uses the Text-to-Pictograph engine to augment emails with sequences of Sclera or Beta pictographs, allowing people with communicative disabilities to familiarize themselves with information technology.

SRILM toolkit (Stolcke, 2002). The second model relies on *word associations* within a broader context: The system identifies the most frequent lemmas in the synset (see section 5.1) of each entered pictograph and retrieves a list of semantically similar words from DISCO,[7] an application that allows to retrieve the semantic similarity between arbitrary words and phrases, along with their similarity scores. Pictographs that are connected to these words are presented to the user.

5 Natural Language Generation from Pictographs

The main challenge in translating from pictograph languages to NL is the fact that a pictograph-for-word correspondence will almost never provide an acceptable output. Pictograph languages often lack pictographs for function words. A single pictograph often encodes information corresponding to multiple words with multiple inflected word forms in NL.

Section 5.1 describes how the bridge between Sclera and Beta pictographs and natural language text was built. The system's general architecture is outlined in section 5.2. It introduces a set of parameters, which were tuned on a training corpus (section 5.3). Finally, as explained in section 5.4, an optimal NL string is selected.

5.1 Linking pictographs to natural language text

Pictographs are connected to NL words through a semantic route and a direct route.

The *semantic route* concerns the use of WordNets, which are a core component of both the Text-to-Pictograph and the Pictograph-to-Text translation systems. For Dutch, we used the Cornetto (Vossen et al., 2008) database. Vandeghinste and Schuurman (2014) manually linked 5710 Sclera and 2746 Beta pictographs to Dutch synsets (groupings of synonymous words) in Cornetto.

The *direct route* contains specific rules for appropriately dealing with pronouns (as pictographs for pronouns exist in Sclera and Beta) and contains one-on-one mappings between pictographs and individual lemmas in a dictionary.

5.2 Architecture of the system

When a pictograph is selected, its synset is retrieved, and from this synset we retrieve all the synonyms it contains. For each of these synonyms, we apply *reverse lemmatization*, i.e. we retrieve the full linguistic paradigm of the lemma, together with its part-of-speech tags. For Dutch, we created a reverse lemmatizer based on the SoNaR corpus.[8]

Each of these surface forms is a hypothesis for the language model, as described in section 5.4. For nouns, we generate additional alternative hypotheses which include an article, based on part-of-speech information.

5.3 Tuning the parameters

The Pictograph-to-Text translation system contains a number of decoding parameters. *Threshold pruning* determines whether a new path should be added to the existing beam, based on the probability of that path compared to the best path. *Histogram pruning* sets the beam width. The *Cost* parameter estimates the cost of the pictographs that still need processing (based on the amount of pictographs that still needs processing). Eventually, *Reverse lemmatizer minimum frequency* sets a threshold on the frequency of a token/part-of-speech/lemma combination in the corpus, limiting the amount of possible linguistic realizations for a particular pictograph. For Dutch, frequencies are based on occurrence within the SoNaR corpus.

These parameters have to be tuned for every pictograph language/NL pair. For Dutch, our tuning set consists of 50 manually translated messages from the WAI-NOT corpus. We ran five trials of local hill climbing on the parameter search space, with random initialization values, in order to maximize BLEU (Papineni et al., 2002). BLEU is a commonly used metric in Statistical Machine Translation. We did this until BLEU converged onto a fixed score. From these trials, we took the optimal parameter settings.

5.4 Decoding

We performed Viterbi-decoding based on a trigram language model, trained with the SRILM toolkit on a very large corpus. The Dutch training corpus consists of Europarl (Koehn, 2005), CGN (Oostdijk et al., 2003), CLEF (Peters and Braschler, 2001), DGT-TM (Steinberger et al., 2012) and Wikipedia.[9]

[7] http://www.linguatools.de/disco/

[8] http://tst-centrale.org/producten/corpora/sonar-corpus/

[9] http://en.wikipedia.org/wiki/

6 Preliminary results

We present results for Sclera-to-Dutch and Beta-to-Dutch. The test set consists of 50 Dutch messages (975 words) that have been sent with the WAI-NOT email system and which were manually translated into pictographs (724 Sclera pictographs and 746 Beta pictographs).[10] We have evaluated several experimental conditions, progressively activating more features of the system.

The first condition is the *baseline*, in which the system output equals the Dutch pictograph names.[11] The next condition applies *reverse lemmatization*, allowing the system to generate alternative forms of the Dutch pictograph names.[12] We then added the *direct route*, which mostly influences pronoun treatment. The following condition adds the *semantic route*, using Cornetto synsets, allowing us to retrieve all word forms that are connected to the same synset as the pictograph. Finally, we let the system generate alternative hypotheses which also include *articles*.

Table 1 shows the respective BLEU, NIST (Doddington, 2002), and Word Error Rate (WER) scores for the translation of messages into Sclera and into Beta. We use these metrics to present improvements over the baseline. As the system translates from a poor pictograph language (with one pictograph corresponding to multiple words and word forms) into a rich NL, these scores are not absolute.[13] Future work will consist of evaluating the system with human ratings by our target group.

Condition	BLEU	NIST	WER
Sclera			
Baseline	0.0175	1.5934	76.4535
Rev. lem.	0.0178	1.6852	76.8411
Direct	0.0420	2.2564	66.9574
Synsets	0.0535	2.5426	65.9884
Articles	0.0593	2.8001	67.4419
Beta			
Baseline	0.0518	2.767	70.4457
Rev. lem.	0.0653	3.0553	70.3488
Direct	0.0814	3.3365	63.0814
Synsets	0.0682	3.1417	61.4341
Articles	0.0739	3.4418	63.1783

Table 1: Evaluation of Pictograph-to-Dutch conversion.

7 Conclusion

These first evaluations show that a trigram language model for finding the most likely combination of every pictograph's alternative textual representations is already an improvement over the initial baseline, but there is ample room for improvement in future work.

The Pictograph-to-English and Pictograph-to-Spanish translation systems are currently in development.

It is important to note that we assume that the grammatical structure of pictograph languages resembles and simplifies that of a particular NL. Nevertheless, the users of pictograph languages do not always need to introduce pictographs in the canonical order or could omit some of them. Future work will look into generation-heavy and transfer approaches for Pictograph-to-Text translation. In the generation-heavy approach, the words conveyed by the input pictographs will be considered as a bag of words. All their possible permutations will be evaluated against a language model (Vandeghinste, 2008). In the transfer system, the input sentence will be (semantically) analyzed by a rule-based parser. A number of transfer rules convert the source language sentence structure into the sentence structure of the target language, from which the target language sentence is generated, using language generation rules. Both methods can be combined into a hybrid system.

User tests will reveal how both the static hierarchy of pictographs and the dynamic prediction tools can be improved.

[10] In future work, we will also evaluate pictograph messages that are created by real users. We thank one of the anynomous reviewers for this suggestion.

[11] Note that Beta file names often correspond to Dutch lemmas, while Sclera pictographs usually have more complex names, including numbers to distinguish between alternative pictographs for depicting the same concept. This explains why the Sclera baseline is lower.

[12] The Sclera file names are often too complex to generate variants for the language model.

[13] For instance, the system has no means of knowing whether the user is talking about a *chicken* or a *hen*, or whether the user *eats* or *ate* a pizza.

References

G. Doddington. 2002. Automatic Evaluation of Machine Translation Quality Using N-gram Co-occurrence Statistics. In *HLT-02*, pages 138–145.

A. Finch, K. Tanaka-Ishii, W. Song, and E. Sumita. 2011. picoTrans: Using Pictures as Input for Machine Translation on Mobile Devices. In *IJCAI-11*, pages 2614–2619.

S. Fitrianie and L. Rothkrantz. 2009. Two-Dimensional Visual Language Grammar. In *TSD-09*, pages 573–580.

H. Jing. 1998. Usage of WordNet in Natural Language Generation. In *COLING-ACL-98*.

T. Keskinen, T. Heimonen, M. Turunen, J.P. Rajaniemi, and S. Kauppinen. 2012. SymbolChat: A Flexible Picture-based Communication Platform for Users with Intellectual Disabilities. *Interacting with Computers*, 24(5):374–386.

P. Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *MT Summit-05*, pages 79–86.

F.-H. Liu, L. Gu, Y. Gao, and M. Picheny. 2003. Use of statistical N-gram models in Natural Language Generation for Machine Translation. In *ICASSP-03*, pages 636–639.

J. Munemori, T. Fukada, M. Yatid, T. Nishide, and J. Itou. 2010. Pictograph Chat Communicator III: a Chat System that Embodies Cross-Cultural Communication. In *KES-10*, pages 473–482.

N. Oostdijk, W. Goedertier, F. Van Eynde, L. Boves, J. P. Martens, M. Moortgat, and H. Baayen. 2003. Experiences from the Spoken Dutch Corpus Project. In *LREC-02*, pages 340–347.

K. Papineni, S. Roukos, T. Ward, and W. Zhu. 2002. Evaluation of Machine Translation. In *ACL-02*, pages 311–318.

C. Peters and M. Braschler. 2001. European Research letter: Cross-language System Evaluation: the CLEF Campaigns. *Journal of the American Society for Information Science and Technology*, 52(12):1067–1072.

R. Steinberger, A. Eisele, S. Klocek, S. Pilos, and P. Schlüter. 2012. DGT-TM: A Freely Available Translation Memory in 22 Languages. In *LREC-12*, pages 454–459.

A. Stolcke. 2002. SRIL: An Extensible Language Modeling Toolkit. In *ICSLP-02*.

P. Tuset, J.M. Barbern, P. Cervell-Pastor, and C. Janer. 1995. Designing Messenger Visual, an Instant Messenging Service for Individuals with Cognitive Disability. In *IWAAL-95*, pages 57–64.

P. Vaillant. 1998. Interpretation of Iconic Utterances Based on Contents Representation: Semantic Analysis in the PVI System. *Natural Language Engineering*, 4(1):17–40.

V. Vandeghinste and I. Schuurman. 2014. Linking Pictographs to Synsets: Sclera2Cornetto. In *LREC-14*, pages 3404–3410.

V. Vandeghinste, I. Schuurman, L. Sevens, and F. Van Eynde. Accepted. Translating Text into Pictographs. *Natural Language Engineering*.

V. Vandeghinste. 2008. *A Hybrid Modular Machine Translation System. LoRe-MT: Low Resources Machine Translation*. LOT, Utrecht.

P. Vossen, I. Maks, R. Segers, and H. van der Vliet. 2008. Integrating Lexical Units, Synsets, and Ontology in the Cornetto Database. In *LREC-08*.